Control and Management Architecture for Multipoint Videoconferencing over ATM Networks[*]

C.-H. Yang, T.-C. Hou, C.-C. Wen, H.-J. Yang, and K.-J. Chen
Department of Electrical Engineering
National Chung Cheng University
160, San-Hsing, Ming-Hsiung
Chiayi 621, Taiwan, R.O.C
p8305@eeipx2.ee.ccu.edu.tw   tch@ee.ccu.edu.tw   p8705@eeipx2.ee.ccu.edu.tw
vom@ee1.ee.ccu.edu.tw   ieekjc@ccunix.ccu.edu.tw

Abstract

In this paper we describe a control and management architecture for an ATM-based Multipoint Videoconferecning (AMV) system. The service specification of the AMV system defines control functions that emulate human interactions, management functions that manage the videoconference and improve the service availability, and a web-based interface for easy control and management of the videoconference. To realize the AMV system, the required functional blocks are partitioned and a control and management architecture that contains a centralized server and distributed device managers is designed. After prototyping and then validating the AMV system on both ATM LAN and ATM WAN, we have the following observations and experiences: (i) Reliable multipoint communications are crucial in controlling and managing the multipoint videoconferencing. (ii) Coordination of the multipoint videoconferencing system must be maintained precisely to avoid the service disruption. (iii) The control and management architecture can well support our AMV system.

Keywords

Architecture, control, management, videoconferencing, ATM, WWW

1. Introduction

Most of the traditional videoconferencing systems are deployed on public switched telephone networks (PSTN) and narrowband circuit-switched and packet-switched networks (e.g., N-ISDN, Internet). The ITU-T (International Telecommunication Union—Telecommunications) has developed many standards for audiovisual service [1]. With these standards, several mature videoconferencing systems [2]-[4]

---

[*] This work is supported in part by National Science Council, Taiwan, ROC, under Grant NSC-86-2213-E-194-032.

have been developed to realize applications such as distance learning and videoconferencing. Traditional videoconferencing systems can be classified into two classes: one is the *circuit-based* videoconferencing and the other is the *packet-based* videoconferencing [5]. Because of the unique characteristics of the underlying networks, each videoconferencing system has its own approaches: the *centralized approach* is used for the circuit-based videoconferencing; the *distributed approach* is used for the packet-based videoconferencing. As for the centralized approach, the multipoint videoconferencing is based on point-to-point, fixed bandwidth, and circuit-switched connections. Thus, a *Multipoint Control Unit* (MCU) [6] is required in order to realize multipoint communication in videoconferencing. In contrast, the distributed approach is considered to be appropriate for packet-based videoconferencing. In order to guarantee an acceptable audio/video quality and tolerable latency, error control, jitter control, and rate control are required [7]. Moreover, to realize multipoint videoconferencing, the multicast technology [8] is needed.

In spite of the many traditional videoconferencing systems that exist today, the ATM-based videoconferencing is becoming promising because of the versatile characteristics of ATM networks. The characteristics include *high available bandwidth*, *multicast capability*, *control and management functionalities*, etc. [1]. Thus, videoconferencing over ATM networks is expected to offer high-quality audio/video and advanced control/management functionalities. There are experimental ATM-based videoconferencing systems reported in the literature [9]-[11]. Some report the experiences with the *design* and *implementation* of the videoconferencing testbed, and some discuss the flexible *service-enabling platforms*, also referred to as the *middleware*. However, to exploit the capability/ characteristics of ATM networks for realizing videoconferencing, more issues must be dealt with. These issues may include *video/audio coding and decoding*, *QoS control*, *videoconferencing control/management*, *system architectures*, *system reliability*, etc.

In this paper we address the issues of designing a control and management architecture for an ATM-based multipoint videoconferencing (AMV) system. First we specify the ATM-based videoconferencing service. The service specification defines the *control functions* that emulate human interactions of meetings, the *management functions* that manage videoconferencing and especially improve the service availability, and a *web-based interface* for easy control and management of the videoconference. To realize these functions, the service specification is partitioned into several functional blocks and a control and management architecture that contains a centralized server and distributed device managers is designed for the AMV system. We tested a prototype AMV system first in a campus ATM LAN then in a wide-area ATM network in Taiwan with a conference connecting participants in seven universities in May 1997. During the experiments, we made some observations and acquired some valuable experience. The rest of this paper is organized as follows. Section 2 describes the specification of the AMV system. Section 3 introduces the platforms of the AMV system. Section 4 presents the prototype design of the AMV system. Section 5 summarizes this paper.

2. Specification of the AMV System

ATM-based videoconferencing is expected to offer conferees high-quality audio/ video and versatile service control/management. To envisage how versatile the ATM-based videoconferencing service can be, we specify the services provided by ATM-based videoconferencing as follows.

2.1 Service Specification

A good videoconferencing system should emulate a face-to-face conference as closely as possible. Thus, by extracting several important characteristics from the face-to-face conference, we can specify the service requirements of ATM-based videoconferencing:

1. *Continuous presence* [2]: In a face-to-face conference, each conferee can control the receipt and assimilation of multimedia information. Thus, a conferee in a videoconference is expected to have free access to any available audio and video.
2. *Conference control*: There are rules to define how a face-to-face conference starts and terminates as well as how the conferees join and leave the conference. We interpret these rules as the conference control of a videoconference.
3. *Floor control*: In conferences, the floor concept is designed to fairly offer participants the opportunities for speaking in the conference and to well conduct the proceeding of the conference. The floor control for videoconferencing is defined to emulate the floor concept and the transfer of the floor.
4. *Status report*: Each conferee in a conference naturally has his/her specific status information which may include "*state*" (e.g., present or absent), "*duty*" (e.g., chairperson, speaker, or listener), etc. Such status information can be implicitly recognized by other conferees through observing conferees' seats, behavior, etc. However, in a videoconference, the conferees sit in remote meeting rooms, and the status information can not be easily obtained. Thus a management function, *status report*, is needed to be accessible to the conferees in the videoconference.
5. *Failure recovery*: A face-to-face conference can successfully proceed without failures because the audio/video and request/response can be distributed and transmitted reliably in a meeting room. However, the videoconferencing service may be disrupted due to the network congestion and/or failures. Thus, failure recovery is needed in videoconferencing to emulate the reliable conference.

The above five elementary services are defined in the specification of our AMV system. Obviously the continuous presence is the key service of the videoconferencing. The conference control and the floor control constitute the fundamental service controls, and both the status report and the failure recovery belong to the service management. Moreover, a user interface is required for conferees to control and manage the videoconference easily.

2.2 Functional Blocks

In ATM networks the unicast/multicast virtual channels (VCs) can be used for the transmission/distribution of high-quality audio and video. Moreover, control and

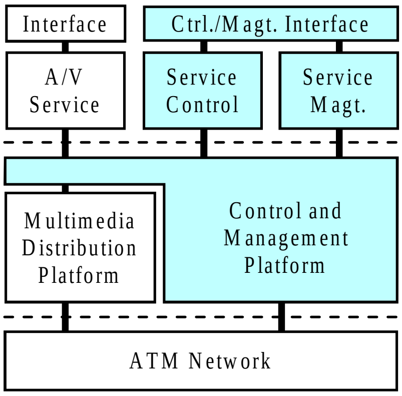


Figure 1: Functional block diagram.

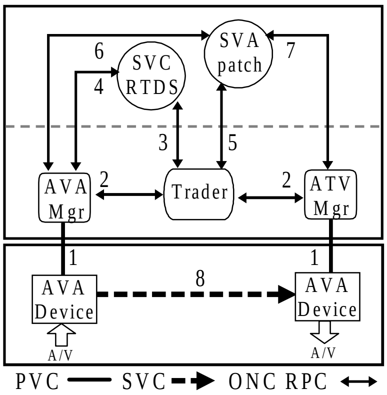


Figure 2: SVA system.

management in the application level can make use of unicast/multicast channels to distribute/gather the commands/responses. In addition, the ATM OAM flows, F4 and F5, can support fault and performance management in both VP (Virtual Path) and VC levels. Thus ATM networks have built-in functionalities to support multimedia distribution as well as control/management of multimedia service.

The functional block diagram for an AMV system is shown in Figure 1. On top of the ATM network are the *Multimedia Distribution Platform* (MDP) and the *Control and Management Platform* (CMP). The MDP is employed to distribute high-quality audio and video by making use of ATM unicast/multicast channels. For example, the SVA system [12] can be used as the MDP. The CMP is a platform that controls and manages the MDP and supports the modules above CMP to realize the audio/video service, service control and service management. To control and manage the MDP, CMP can make use of the channels and management functionalities provided by the underlying ATM network. Above the CMP are the audio/video service module, the service control module, and the service management module. The audio/video service module is capable of audio/video processing and can provide participants with continuous presence. The service control module and the service management module are designed to realize the conference control, floor control, status report, and failure recovery. Moreover, two user interfaces are required to provide participants with easy ways to control and manage the conference as well as to select the audio and video, respectively.

As shown in Figure 1, three levels in the functional block diagram can be distinguished: *network*, *platform*, and *service*. For our AMV system we naturally select an ATM network in the lower level. In the middle level, the SVA system is selected as the MDP. Since SVA is capable of audio/video processing, continuous presence can also be realized by employing the SVA system. Aside from the MDP and the ATM network, we have developed CMP, the service control module, the service management module, and the control and management interface for the AMV system. In the following section, we introduce the SVA system and present the design of the CMP.

3. Platforms of the AMV System

3.1 SVA System

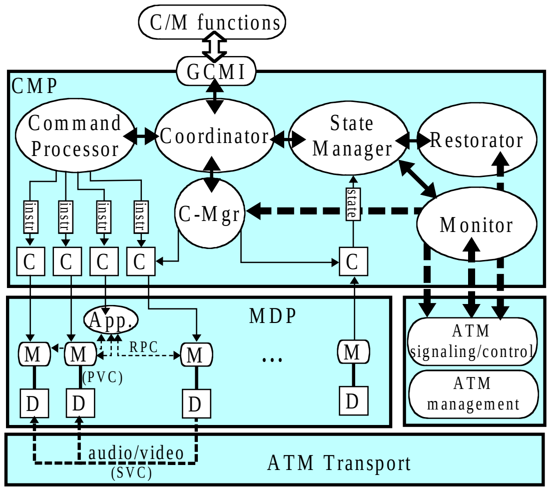
An SVA system contains two hardware devices, Audio Video Adapter (AVA) and ATM TV (ATV). An AVA device can receive analog audio/video signals from microphone/camera, and then compress the received signals to a Motion-JPEG stream. Such a stream is encapsulated in AAL5 PDUs. The AAL5 PDUs then are transmitted to remote ATV device(s) through an ATM switched virtual channel (SVC). At the receiving end, the ATV receives AAL5 PDUs and reconstructs the Motion-JPEG stream. ATV then decompresses the Motion-JPEG stream to analog signals, which can be displayed on TV.

The software in SVA includes managers (i.e., *AVA manager*, *ATV manager*, and *trader*) and applications (i.e., *SVC real-time display software* (*SVC-RTDS*) and *SVA-patch*). Referred to Figure 2, AVA/ATV managers can control AVA/ATV's operations. The trader provides the naming service for applications (e.g., SVC-RTDS). When an AVA/ATV manager is activated, the AVA/ATV manager must register itself in a trader. The SVC-RTDS application, after looking up the intended device managers in trader(s), is capable of specifying the parameters (e.g., frame rate) of an audio/video stream by controlling the source AVA manager. Besides specifying streams, SVC-RTDS can provide users with basic management functions, stream state monitoring, and device state monitoring. Similarly, on the basis of trader's naming service, SVA-patch can create, patch together, and display video/audio streams by controlling source AVA and sink ATV managers.

As for the communication channels, the SVA platform uses ATM PVC for manager-to-device communication and SVCs for transmission of audio/video streams from AVA to ATV. Besides, communication among traders, managers, and applications is through ONC RPC. We show how the SVA system works step by step in Figure 2: 1. setup PVCs, 2. managers register with the trader, 3. application looks up managers in the trader, 4. application specifies streams, 5. SVA-patch looks up managers in the trader, 6. SVA-patch controls the AVA manager, 7. SVA-patch controls the ATV manager, 8. AVA sets up SVC and distributes audio/video.

3.2 Control and Management Platform

As described above, the SVA system is capable of distributing high-quality audio/video over ATM networks and patching multiple video streams together at the receiving end. Nevertheless, the SVA system does not fully support interactive video/audio and advanced service control/management functions defined by our service specification. Thus we design a control and management platform (CMP), which can support us to develop the service control and management functions for videoconferencing. Note that, besides the CMP, a middleware approach [9] is also a possible solution. The middleware approach can provide a standardized interface for various resources in networks, computing platforms, and multimedia devices. With supports from the middleware, a multimedia service can be implemented easily and quickly. However, we design the AMV system with a particular



C: ATM channel M: device manager D: multimedia device
IPC: ⟷ Access to ATM: ⟷ Controls: ⟶

Figure 3: Structure of the CMP.

emphasis on the system reliability and service availability. The middleware approach does not provide a direct way to improve the system reliability. In [13], the authors address why constructing highly available applications with CORBA is difficult.

Referred to Figure 3, the CMP consists of the *General Control and Management Interface (GCMI)*, *coordinator*, *command processor*, *state manager*, *monitor*, *restorator*, *ATM channels*, and *channel manager*. Above the CMP are control and management functions that are to realize the conference control, floor control, status report, and failure recovery; beneath the CMP are the MDP and the ATM network. The MDP consists of several multimedia devices, device managers (for short, D-Mgr), and applications (e.g., SVA-patch). To control and manage the MDP easily, the CMP bundles up tedious SVA instructions and makes use of ATM management functionalities to provide eight operations to be invoked by the control and management functions. The operations are 1. `CAVS`: *Create a point-to-point/point-to-multipoint Audio/Video Stream*, 2. `TAVS`: *Terminate an Audio/Video Stream*, 3. `XAS`: *miX Audio Streams*, 4. `XVS`: *patch (miX) Video Streams*, 5. `MSS`: *Monitor the State of a Stream*, 6. `RFS`: *Restore the Failed Stream*, 7. `MSDM`: *Monitor the State of multimedia Device Manager*, 8. `RFDM`: *Recover the Failed multimedia Device Manager*.

We briefly discuss how the CMP works. When a conferee joins a conference, the conference control invokes `CAVS` to create a new video stream and invokes `MSS` and `MSDM` to monitor the stream state and the device manager state, respectively. Referred to Figure 3, the above invocations are passed to the *coordinator* through *GCMI*. The *coordinator* then activates and coordinates *command processor*, *state manager*, and *channel manager* (*C-Mgr*). First, to

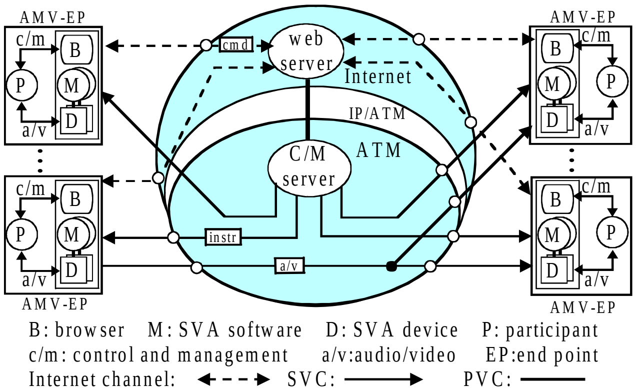


Figure 4: Architecture of the AMV system.

execute `CAVS`, the *coordinator* activates the *command processor* to generate SVA instructions, and also activates *C-Mgr* to setup a new channel for distributing the instructions to *D-Mgrs*. The *D-Mgrs* receive and execute the instructions to create new video streams. After creating the video stream, the *coordinator* activates the *state manager* to execute `MSS` and `MSDM`. To execute `MSS`, the *state manager* invokes the *monitor* to perform, by making use of OAM flows, fault/performance management for the VC that is used for transmission of the video data. To execute `MSDM`, the *C-Mgr* sets up a channel for the transmission of *D-Mgr* state from the *D-Mgr* to the *state manager*. After finishing these operations, the new participant joins the conference at the platform level. Similarly, floor control can invoke proper operations to achieve the corresponding control and management of the MDP.

As for the management functions, *status report* is performed periodically and *failure recovery* is activated when a failed VC or abnormal state of *D-Mgr* is detected. When the *state manager* wants to get the state of a *D-Mgr*, the *state manager* activates the *C-Mgr* through the *coordinator* to obtain a channel. Thereafter, through the channel, the *state manager* can get the state of *D-Mgr*. Furthermore, both the *monitor* and the *restorator* are responsible for failure recovery by making use of ATM management and signaling/control.

4. Prototype Design of the AMV System

4.1 Architecture of the AMV System

Two alternatives of architecture can be employed to realize the CMP and control/management functions: *centralized architecture* and *distributed architecture*. For the distributed architecture, the software modules that realize the CMP and control/management functions are running at different computers. On the other hand, for the centralized approach, the software modules are running at a computer that is referred to as the *control and management server*. Obviously, software modules in the distributed architecture involve network communication, and some critical issues, such as coordination of the distributed modules and

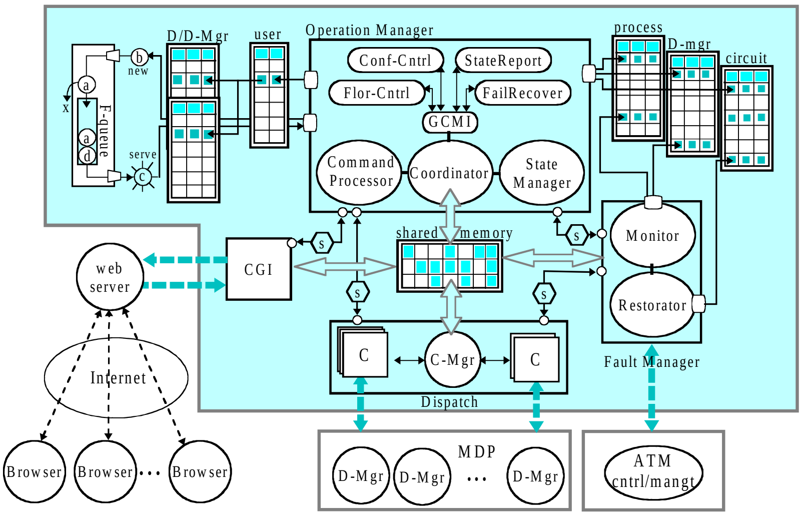


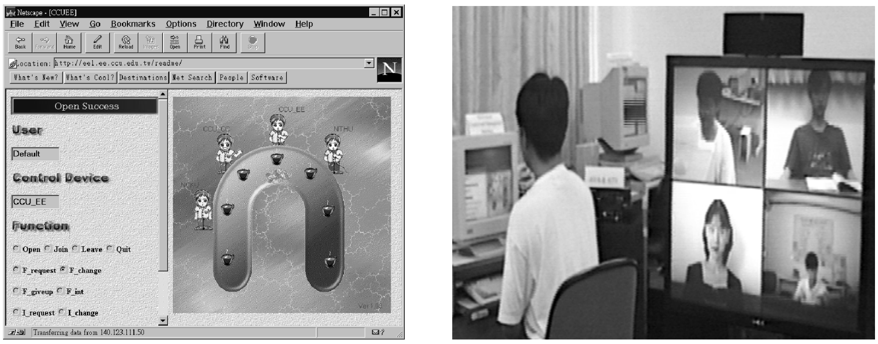
Figure 5: Software modules of the control and management server.

synchronization of the tables for control and management, have to be carefully dealt with. For interactive audio-visual service, many interactions are involved and the coordination and synchronization become crucial in maintaining the system reliability. On the other hand, instead of network communication, only IPC (inter-process communication) is required in the centralized architecture. And the coordination and synchronization can be dealt with more easily. Nevertheless, the risk of the centralized architecture is that crash of the server will cause the disruption of the videoconferencing service. However, to recover the crash of the centralized server is easier than to recover the failed distributed system that involves many network communications and introduces complex coordination and synchronization problems. Thus, we select the centralized architecture to realize the AMV system.

Besides the MDP and the control and management server, we define a user-friendly interface for participants to control and manage the videoconference. The interface has to provide all remote participants with an easy way to control and manage the videoconference. A popular approach is the web-based interface. The architecture consisting of browsers and a web-server is employed to realize the web-based interface. As for the function of the continuous presence, SVA-patch can be used for video composition. An infra-red control that comes with the ATV device is used to select the intended video streams that are sunk at the ATV device. Thus we have selected a proper architecture for the AMV system, and we show the architecture in Figure 4.

4.2 Software Modules of the Control and Management Server

Referred to Figure 5, the server consists of four main modules: *operation manager*, *fault manager*, *dispatch*, and *CGI*. A *shared memory* is employed for communications among the four modules. In order to synchronize the data in the



Open Success

User

Default

Control Device

CCU_EE

Function

○ Open ○ Join ○ Leave ○ Quit

○ F_request ● F_change

○ F_giveup ○ F_int

○ I_request ○ I_change

Transferring data from 140.123.111.50

(a) (b)

Figure 6: The interface and operations of the AMV system.

shared memory, four *semaphores* are used to control the access to the shared memory. Besides, several tables and a floor queue are designed for the *operation manager* and the *fault manager*. Outside the server, there are MDP, ATM networks, and a *web server* with *browsers*. We explain how the control/management server works as follows.

- Control and management interface: Users can initiate a command by clicking on the corresponding icon of the user interface. The command then is sent from the *browser* to the *web server* through the Internet. *Web server* receives the command and then invokes *CGI* to execute the command. To execute the command, *CGI* activates the corresponding control/management function(s) in the *operation manager*. The communication between *CGI* and the *operation manager* makes use of the shared memory and the semaphore. After the *operation manager* executes the corresponding functions, the response is sent to *CGI* via the shared memory. *CGI* then responds to the user via the *web server*, the Internet channel and the *browser*. The user interface provides participants with a graphic user interface (GUI). Two control functions are *conference control* and *floor control*. The commands include "*open*", "*join*", "*leave*", "*quit*", "*floor-request*", "*floor-change*", "*floor-giveup*", etc. As for the status report, the GUI can show the participant icons around a meeting table. Different icons imply different status of the participant. The status contains duty (chairman, speaker, or listener), state (current floor holder, next floor holder, request floor, none). We show the user interface of control and management in Figure 6-(a).
- Operation manager: The *operation manager* consists of the four control and management functions at the service level as well as *GCMI*, *coordinator*, *command processor*, and *state manager* at the CMP level. We explain how the control and management functions work with CMP components as follows:
  a) Conference control: The conference control maintains the information of the participants that is recorded in the user table. An entry in the user table is also associated with the multimedia devices and the device managers that are kept in the device table and the device manager table, respectively. All

entries in the table are filled when the system operator initializes the AMV system. The conference control is responsible for the management of the participants' status. For example, when participant A opens a video-conferencing, then the conference control has to change the status of participant A into "*chairman*" and "*active*". Afterwards, if participant B joins the videoconference, the conference control changes B's status into "*conferee*" and "*active*." On the other hand, when participant B leaves, B's status has to be changed from "*active*" to "*initial*" by the conference control. Besides managing the information of the participants in the function level, the conference control also invokes the corresponding CMP operations. For example, when participant B joins the videoconferencing, the conference control has to invoke `MSDM` at least to verify the active states of the device managers that are associated with participant B.

b) Floor control: In a videoconference, a participant can request the floor. Such a floor request is sent from the interface to the floor control. In order to maintain the sequence of the floor requests, a FIFO queue without duplicate entries is designed for the floor control. Then the floor control can manage the floor requests and select a floor request to serve. For example, when a floor request from participant C becomes the oldest, participant C can have the floor for a pre-defined *floor-holding-time*. Then floor control has to invoke CMP operations to rearrange the video distribution and/or video composition as well as audio distribution and/or mixing.

c) Status report: The status report is responsible for managing the association between the user table and device/manager tables. The status report can periodically invoke CMP operations to monitor the devices and managers. If the device(s) and/or manager(s) is (are) down for a pre-defined time interval, the status report changes the participant status into "*inactive*." Such a change of the participant status is revealed in the interface. Besides the upward reporting, the status report is also responsible for the update of the participant's status at the user level. For example, when participant C gets the floor, his/her status will be changed from a listener to a speaker. Such a status change must be reported to all participants' interfaces by the status report.

d) Failure recovery: The failure recovery function is realized through the aid from the *fault manager*. The *fault manager* performs the failure recovery on three types of the objects: VCs, audio/video streams, and processes. The *monitor* in the *fault manager* periodically queries the state of each object in the AMV system. When the *monitor* detects a failed object, the *restorator* in the *fault manager* is activated to recover the object. After the *restorator* recovers the object, the *monitor* continues monitoring the object. Otherwise, the *restorator* reports the failed object to the failure recovery function. The failure recovery function will reveal the failure status of a conferee through the control/management interface when the object(s), which belong(s) to the conferee, fail(s).

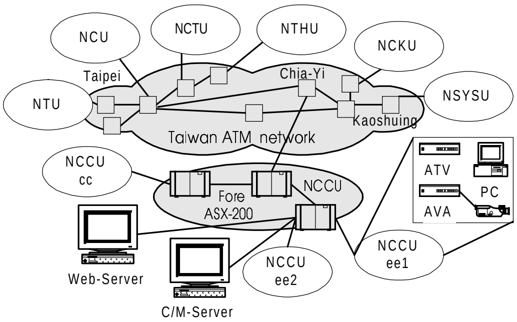


Figure 7: Test environment.

- Dispatch: The *dispatch* consists of a *channel manager* (*C-Mgr*) and a number of ATM channels for control and management. The *channel manager* is responsible for the setup and tear-down of the ATM channels. Besides, the *C-Mgr* feeds commands to the channels as well as collects the information from the channels. Such commands and information are stored in the shared memory. The *operation manager* and *fault manager* make use of the semaphores to inform the *dispatch* to send commands or receive information.

As discussed above, the control and management server accomplishes the services described in the service specification. Figure 6-(b) shows a snapshot of the operations of the AMV system.

4.3 Experimental Results

After implementing the AMV system, we tested it on a campus ATM LAN and a wide-area ATM network in Taiwan. Thereafter, we held a videoconference having participants from seven universities in Taiwan. Figure 7 shows the experimental ATM network that is deployed over Taiwan. Nine locations, each one equipped with AVA, ATV, PC, camera, etc., are connected to the ATM network. From the tests, we have some experimental results. We discuss the results as follows.

- Tests on ATM LAN: As shown in Figure 7, the campus ATM network consists of three nodes; two workstations, one SUN SPARC 20 with 64 MB RAM and one SUN SPARC 5 with 64 MB RAM, are employed to be the control and management server and the web-server, respectively. We experiment the AMV system on the campus ATM network successfully; however, the workstation in which the web server is running has heavy loading during the tests. All of the functions are tested, and the functions can provide the expected services for participants. The web-based interface does provide an easy way for control and management of videoconferencing, and it can support nearly real-time control and management of videoconferencing in LAN environment.
- Tests on ATM WAN: After testing the AMV system on a campus ATM network successfully, we continue to test it on the ATM WAN. The following experimental results are observed:

1. Multicast of audio and video information can run smoothly in ATM WAN environment. The highly interactive high-quality audio/video involved in the multipoint videoconference can be accomplished by operating the AMV system not only in ATM LAN environment but also in ATM WAN environment. Moreover, the AMV system can support up to nine participants to attend the videoconference during the experiment. The control and management functions can provide the expected services. However, two problems are found during the experiment (item 2. and 3. below).
2. Due to the fluctuating packet delay in the Internet (i.e., WAN environment), the control and management interface may not respond to the participant in real time. The participants then click the control icon to generate a large number of duplicate commands that further stress the web server.
3. When congestion occurs in the wide-area ATM network, the ATM channels used by the control and management server to distribute instructions to the device managers become unreliable. Though the error control can resolve the PDU loss, the delay of distributing the instructions to the different device managers will fluctuate. Besides, the SVA system sometimes reveals the following error message, *ONC RPC fail (time-out)*, when SVA runs in the inter-networking environment (e.g., IP over ATM). The ONC RPC failure can cause the SVA to fail in supporting CMP to achieve service control and management. When both the network congestion and the ONC RPC failure occur, the coordination of the AMV system may be violated temporarily.

In summary, the AMV system operates successfully in the LAN and WAN environments. However, in the WAN environment, the web-based interface may not respond to participants in real time and the coordination of the AMV system may be violated temporarily due to the fluctuating packet delay, network congestion, and ONC RPC failure. To provide a highly reliable multipoint videoconferencing service, the above issues must be resolved. First, instead of the web-based interface, the ATM-based interface may be more appropriate for real-time control and management. Second, to resolve the complex coordination issue, a reliable multipoint control protocol is required [14]. With the protocol, the order of the instructions distributed to the device managers can be maintained, and then the CMP can control MDP precisely. Third, to deal with the RPC failure in the inter-networking environment is a complicated problem [15] for future study.

5. Summary

In this paper we address a control and management architecture of an ATM-based multipoint videoconferencing (AMV) system. First, to envisage how versatile the ATM-based videoconferencing service can be, we define five elementary services for the AMV system; these services are *continuous presence*, *conference control*, *floor control*, *status report*, and *failure recovery*. Then, we depict the functional blocks of the AMV system. Two important functional blocks are

Multimedia Distribution Platform (*MDP*) and *Control and Management Platform* (*CMP*). A commercial SVA system that contains the multimedia devices and the software can be employed as the MDP. We design the CMP, which can be used to develop the service control and management functions. Moreover, a centralized architecture is adopted in the prototype AMV system. The architecture contains a web server, distributed web browsers, a control and management server, and distributed multimedia devices and device managers. After prototyping the AMV system, we validated the functions of the AMV system in a campus ATM network successfully. We then tested the AMV system in a wide-area ATM network in Taiwan with a conference connecting participants in seven universities in May 1997. During the tests, we had some observations and experiences. i) The web-based interface provides an easy way to control and manage the videoconference, but may not fully support real-time control and management of the videoconference operating in a WAN environment. ii) Reliable multipoint communication is crucial in controlling and managing the multipoint videoconferencing. iii) With the delicate multipoint control and management, the coordination of the AMV system can be maintained more easily. iv) To effectively operate the highly interactive ATM-based videoconferencing in the WAN environment, several issues, such as multipoint control/management, coordination of distributed system, etc., have to be resolved.

Acknowledgement

The authors would like to thank the members of Computer Center in National Chung Cheng University, Taiwan. With their cooperation we successfully run the AMV project and the videoconference trials.

References

[1] S. Okubo, S. Dunstan, G. Morrison, M. Nilsson, H. Radha, D. L. Skran, and G. Thom, "ITU-T standardization of audiovisual communication systems in ATM and LAN environments," *IEEE J. on Selected Areas in Commu.*, vol. 15, no. 6, pp. 965-981, Aug. 1997.

[2] M. E. Lukacs and D. G. Boyer, "A universal broadband multipoint teleconferencing service for the 21st century," *IEEE Commu. Mag.*, Nov. 1995, pp. 36-43.

[3] W. J. Clark, "Multipoint multimedia conferencing," *IEEE Commun. Magazine*, pp. 44-50, May. 1992.

[4] S. R. Ahuja and J. R. Ensor, "Coordination and control of multimedia conferencing," *IEEE Commun. Magazine*, pp. 38-43, May. 1992.

[5] M. H. Willebeek-LeMair and Z.-Y. Shae, "Videoconferencing over packet-based networks," *IEEE J. on Selected Areas in Commu.*, vol. 15, no. 6, pp. 1101-1114, Aug. 1997.

[6] M. H. Willebeek-LeMair, D. D. Kandlur, and Z.-Y. Shae, "On multipoint control units for videoconferencing," in *Proc. 19th Conf. on Local Computer Networks*, Oct. 1994.

[7] T. Turletti and C. Huitema, "Videoconferencing on the Internet,"

IEEE/ACM Trans. Networking, vol. 4, no. 3, pp. 340-351, Jun. 1996.

[8] M. R. Macedonia and D. P. Brutzman, "Mbone provides audio and video across the Internet," *Computer*, Apr. 1994.

[9] W. S. Choe, T. J. Geok, W. W. Guo, and T. S. Woon, "ATM-based multi-party conferencing system," in *Proc. GLOBECOM '95*, Nov. 1995, pp. 592-6.

[10] D. Saha, D. Kandlur, T. Barzilai, Z. Shae, and M. Willebeek-LeMair, "A video conferencing testbed on ATM: design, implementation and optimizations", in *Proc. IEEE Int. Conf. on Multimedia Computing and Systems*, May 1995.

[11] K. Smith and R. Pretty, "ATM RendezView: multipoint conferencing on ATM," in *Proc. IEEE Int. Conf. on Multimedia Computing and Systems*, Jun. 1997.

[12] Nemesys Research Limited, "SVA 2.0 user manual," April 1996.

[13] S. Maffeis and D. C. Schmidt, "Constructing reliable distributed communication systems with CORBA," *IEEE Commun. Magazine*, pp. 56-60, Feb. 1997.

[14] C. Diot, W. Dabbous, and J. Crowcroft, "Multipoint communication: a survey of protocols, functions, and mechanisms," *IEEE J. on Selected Areas in Commu.*, vol. 15, no. 3, pp. 277-290, Apr. 1997.

[15] G. J. Armitage, "IP multicasting over ATM networks," *IEEE J. on Selected Areas in Commu.*, vol. 15, no. 3, pp. 445-457, Apr. 1997.

Biography

Chorng-Horng Yang is currently pursuing the Ph.D. degree at the Department of Electrical Engineering, National Chung Cheng University, Taiwan. His research interests are application management, system architectures, and protocols.

Ting-Chao Hou received the Ph.D. degree in electrical engineering from the University of Southern California in 1985. From 1985 to 1993 he was a member of Technical Staff, AT&T Bell Laboratories. He is presently an associate professor in the National Chung Cheng University. His research interests are ATM switching systems, intelligent networks, communication network performance analysis, and wireless networks.

Chao-Cheng Wen is presently a Ph.D. student at the Department of Electrical Engineering, National Chung Cheng University, Taiwan. His research interests are internetworking, IP over ATM, and service availability.

Horng-Jang Yang received the M.S. degree from the National Chung Cheng University in 1996. He joined the AMV project during his studying at the National Chung Cheng University. He currently serves in the Army of ROC.

Kim-Joan Chen received the Ph.D. degrees in applied mathematics from the State University of New York at Stony Brook in 1983. He is currently a professor at the Department of Electrical Engineering, National Chung Cheng University, Taiwan. From 1993 to 1997 he served as the Director of Computer Center and conducted the AMV project. His research interests are multimedia systems, video on demand, video coding, and protocols.